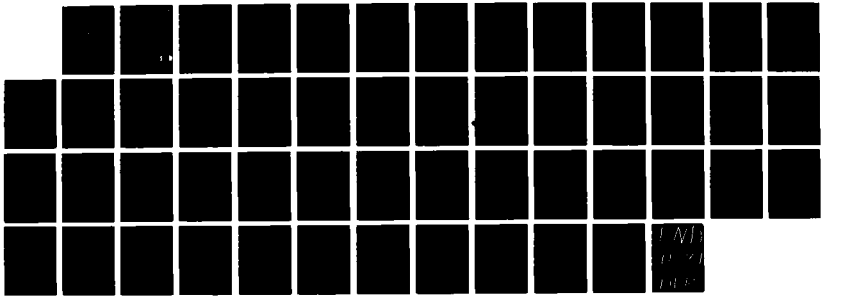
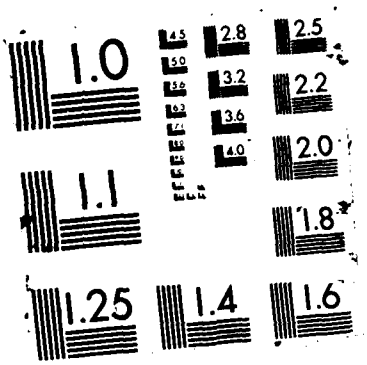
AD-A185 503 EVENT-BASED ESTIMATION OF INTERACTING MARKOV CHAINS WITH APPLICATIONS TO..(U) MASSACHUSETTS INST OF TECH CAMBRIDGE LAB FOR INFORMATION AND D.. 1/1

UNCLASSIFIED P C DOERSCHUK ET AL. SEP 86 LIDS-P-1611 F/G 6/5 NL




1.0
1.1
1.25
1.4
1.6
1.8
2.0
2.2
2.5
2.8
3.2
3.6
4.0
4.5
5.0
5.6
6.3
7.1
8.0
9.0

Unclassified

SECUR

DTIC FILE COPY

AD-A185 583

DOCUMENTATION PAGE

1a. RE Unc1	1b. RESTRICTIVE MARKINGS
2a. SECURITY CLASSIFICATION AUTHORITY	3. DISTRIBUTION/AVAILABILITY OF REPORT: Approved for public release; distribution unlimited
2b. DECLASSIFICATION/DOWNGRADING SCHEDULE	
4. PERFORMING ORGANIZATION REPORT NUMBER(S)	5. MONITORING ORGANIZATION REPORT NUMBER(S): AFOSR-TR- 87-1051
6a. NAME OF PERFORMING ORGANIZATION: Massachusetts Inst. of Tech.	6b. OFFICE SYMBOL (If applicable):
6c. ADDRESS (City, State and ZIP Code): Room 35-311, Cambridge, MA 02139	7a. NAME OF MONITORING ORGANIZATION: AFOSR/NM
8a. NAME OF FUNDING/SPONSORING ORGANIZATION: AFOSR	8b. OFFICE SYMBOL (If applicable): NM
8c. ADDRESS (City, State and ZIP Code): Bldg 410, Bolling AFB DC 20332-6448	7b. ADDRESS (City, State and ZIP Code): Bldg 410, Bolling AFB DC 20332-6448
	9. PROCUREMENT INSTRUMENT IDENTIFICATION NUMBER: AFOSR-82-0258
	10. SOURCE OF FUNDING NOS.

PROGRAM ELEMENT NO.	PROJECT NO.	TASK NO.	WORK UNIT NO.
61102F	2304	A5	

11. TITLE (Include Security Classification)

12. PERSONAL AUTHOR(S): Professor Peter C. Doerschuk

13a. TYPE OF REPORT	13b. TIME COVERED	14. DATE OF REPORT (Yr., Mo., Day)	15. PAGE COUNT
Reprint	FROM _____ TO _____		

16. SUPPLEMENTARY NOTATION

17. COSATI CODES		
FIELD	GROUP	SUB. GR.

18. SUBJECT TERMS (Continue on reverse if necessary and identify by block number)

19. ABSTRACT (Continue on reverse if necessary and identify by block number)

In this paper we examine the problem of estimating the state of a distributed finite-state Markov process consisting of several interacting finite-state systems each of whose transition probabilities are influenced by the states of the other processes. The observations on which the estimation procedure is based are continuous signals containing signatures indicative of the occurrence of particular events in the various finite-state systems. The problem of electrocardiogram analysis serves both as the primary motivation for this investigation and as the source of a case study we describe in the paper. The principal focus of the paper is on the development of an approach that overcomes the combinatorial explosion of truly optimal estimation algorithms. We accomplish this by constructing a systematic design methodology in which the resulting estimator consists of several interacting estimators, each focusing on a particular subprocess. Important questions that we address concern the way in which these estimators interact and the method each estimator uses to account in its own model for the influence of other subprocesses.

20. DISTRIBUTION/AVAILABILITY OF ABSTRACT	21. ABSTRACT SECURITY CLASSIFICATION
UNCLASSIFIED/UNLIMITED ☐ SAME AS RPT. ☐ DTIC USERS ☐	Unclassified

22a. NAME OF RESPONSIBLE INDIVIDUAL	22b TELEPHONE NUMBER (Include Area Code)	22c. OFFICE SYMBOL
Major Woodruff	(202) 767-5026	NM

DD FORM 1473, 83 APR EDITION OF 1 JAN 73 IS OBSOLETE.

SECURITY CLASSIFICATION OF THIS PAGE

September 1986 LIDS-P-1611

Event-Based Estimation of Interacting Markov Chains with Applications to Electrocardiogram Analysis*

Peter C. Doerschuk**

Robert R. Tenney†

Alan S. Willsky‡

Abstract

In this paper we examine the problem of estimating the state of a distributed finite-state Markov process consisting of several interacting finite-state systems each of whose transition probabilities are influenced by the states of the other processes. The observations on which the estimation procedure is based are continuous signals containing signatures indicative of the occurrence of particular events in the various finite-state systems. The problem of electrocardiogram analysis serves both as the primary motivation for this investigation and as the source of a case study we describe in the paper. The principal focus of the paper is on the development of an approach that overcomes the combinatorial explosion of truly optimal estimation algorithms. We accomplish this by constructing a systematic design methodology in which the resulting estimator consists of several interacting estimators, each focusing on a particular subprocess. Important questions that we address concern the way in which these estimators interact and the method each estimator uses to account in its own model for the influence of other subprocesses.

---

*The research described in this paper was supported in part by the Air Force Office of Scientific Research under Grant AFOSR-82-0258. The first author was supported by fellowships from the Fannie and John Hertz Foundation and the M.D.-Ph.D. Program at Harvard University (funded in part by Public Health Service, National Research Award 2T 32 GM07753-06 from the National Institute of General Medical Science).

**Laboratory for Information and Decision Systems, M.I.T., Cambridge, MA, 02139 and Harvard Medical School, 25 Shattuck St., Boston, MA. 02115.

†Laboratory for Information and Decision Systems, M.I.T., Cambridge, MA, 02139 and Alphatech, Inc., 2 Burlington Executive Center, 111 Middlesex Tpke., Burlington, MA, 01803

‡Laboratory for Information and Decision Systems, M.I.T., Cambridge, MA, 02139.

I. Introduction

In Doerschuk (1986) we developed a methodology for modeling electrocardiograms (ECG's) that could be used as the basis for ECG signal processing/analysis algorithms. The models we have developed have several important characteristics:

(1) The models are hierarchical in nature. Specifically, at the upper level we model the event structure of the heart, capturing the electrical state of various anatomical portions of the heart. At the lower level we model the actual observation, the ECG signal. The model reflects the fact that particular cardiac events directly result in the various waveforms seen in the ECG by having particular changes of state in the upper model initiate waveforms in the lower level model.

(2) The upper level model consists of interacting finite-state processes, each of which models a specific anatomical portion of the heart. In this way we attempt to capture the distributed but coordinated way in which the heart operates. In particular, this model structure allows us to highlight the aspects of timing and control that are critical to cardiac behavior. Specifically, in our models the state transition probabilities of each subprocess are affected by the states of other subprocesses, allowing us to model, for example, the attempt of one portion of the heart to precipitate an event (e.g. a contraction) in another portion of the heart.

For a detailed description of this modeling methodology see [Doerschuk 1985, 1986]. An example is also described in the next section as we develop our approach to estimation.

The motivation for developing models of this type was to overcome limitations of existing signal processing models (e.g. those used by Gustafson 1978a, b, 1981; Ciocloda 1983; Gersch 1970, 1975; Tsui 1975; Grove 1978; Haywood 1977; Richardson 1976) by capturing physiology in a more fundamental way while, however, stopping far short of the level of detail found in

physiologically accurate models. We have attempted to develop models that capture all features needed to characterize differences in cardiac rhythms, i.e. differences in the sequential nature of various cardiac events. Consequently, the focus of our interest in this paper is on the estimation of the states of the upper level of the model. As discussed in [Doerschuk 1986], previously developed rhythm analysis methods have also been based on event-level models of cardiac behavior. These models differ from ours in several important respects. Specifically, in previous methods typically only the event level description is modeled, and it is assumed that event-level inputs are available from a wave detection preprocessor. For methods in which attention is paid only to ventricular events -- i.e. to so-called R-waves -- extremely useful models of this type can be developed since (a) R-waves are comparatively large and thus can be detected with great reliability; and (b) one can achieve great efficiencies in describing event-level models by using as a sequential index the successive R-wave occurrences rather than the real ECG sample-to-sample time. However, when one begins to include more cardiac detail, such as the P-waves arising from atrial activity, difficulties arise with such high-level models. In particular, because of their much lower energy levels P-waves are much more difficult to detect. Consequently, in some previous methods one finds ad hoc attempts at feedback from the rhythm tracker to the wave detection process, taking advantage of the fact that in a normal heartbeat the P-wave precedes the R-wave in a predictable fashion. Furthermore, it is very difficult in such a framework to incorporate the possibility of missed detections or false alarms, which are very real

possibilities, especially for P-waves and especially in aberrant cardiac rhythms. For our two-level models in which the measurements are the raw data, wave detection feedback and the possibilities of false positive and missed detections are built in in a fundamental way. Also, while in a perfectly normal rhythm P-waves and R-waves are always paired, cardiac arrhythmias are characterized by asynchronous behavior, in which for example several P-waves may occur without an intervening R-wave or vice versa. The emphasis in our models on timing captures such asynchronous behavior in a natural way rather than in the less than completely satisfactory manner found in previous methods.

The various potential advantages we have attributed to using the models in [Doerschuk 1986] for ECG rhythm processing do not come, of course, without a significant price. In particular, while we do contend that truly optimal estimation based on these models would achieve these advantages, the computational load associated with optimal processing is prohibitively large. Thus the major issue is the development of feasible, suboptimal estimation algorithms. In this paper we investigate the development of such algorithms that take advantage of two important features of this class of estimation problems. First, the estimation of event sequences in the upper level model is essentially a decoding problem (i.e. the ECG is an encoding of the discrete cardiac events we wish to estimate). Consequently we make repeated use of an efficient technique for optimal estimation of finite-state processes first developed for coding applications, namely the Viterbi algorithm [Forney 1973]. Second, since our models are distributed, we can consider the design of distributed estimators, consisting of interacting algorithms each focused on the job of estimating the state of a particular subprocess. Such estimation

structures offer the attractive possibility of implementation in a distributed processor, thereby allowing significant improvements in throughput rates.

The design of such estimators also raises a number of important questions. In particular, since the several subprocesses of our upper level model interact strongly, it is not possible to estimate the state of a subprocess without accounting for the influence on it of other subprocesses. Consequently it is necessary to include an (hopefully highly) aggregated model of other subprocesses that captures the dynamics of the interactions these subprocesses have with the particular subprocess being estimated. Also, it is necessary for the estimators of interacting subprocesses to interact themselves (e.g. estimators of atrial and ventricular activity most certainly have information worth sharing!). This raises an interesting problem. Specifically, since each estimator in essence receives additional measurements in the form of information passed from other estimators, it is necessary to model these measurements -- i.e. each estimator needs an aggregated model of the dynamics and the uncertainties in the other estimators. In addition, since each estimator is using the same raw data (the ECG) but is interested in only some of the events in the data, it may be necessary to provide information to each estimator concerning estimated times of occurrence of other events in the ECG data (e.g. an atrial estimator may need estimates of R-wave locations from the ventricular estimator in order to assist it in locating the much smaller P-waves). Also, as one might expect, there may very well be a need for some iteration in this process so that a high level of performance and consistency among the estimators is achieved.

While electrocardiogram analysis has provided the motivation and examples

for our work, there are a variety of other applications in which similar estimation problems arise. In particular, consider the distributed monitoring of interconnected power systems. Such systems are made up of strongly interacting components subject to events (such as generator trips and line faults) that can precipitate events in other parts of the system. An extremely important problem is the design of distributed monitoring systems, and a critical aspect of this problem is determining how to structure the interaction among local monitoring systems in order to produce a consistent and accurate overall estimate of system status. Similar issues also arise in military contexts in distributed battle management and assessment. In all of these problems the key question is how do we design distributed event estimation algorithms, and in this paper we address this problem and in particular the several critical issues raised in the previous paragraph. We begin in the next section with a case study, in which we describe the details of designing an estimator for an example corresponding to a particular cardiac arrhythmia. This case study allows us to introduce the major questions that arise in designing distributed event estimation algorithms. In Section 3 we then step back from this example and extract from it a general, systematic design approach. We then discuss a series of simplifications that are possible for the class of models arising in ECG analysis and that result in the algorithm described in Section 2.

While the motivation for considering distributed estimators for ECG rhythm analysis is the issue of computational complexity (as opposed to geographic separation of sensors and controls as in the case of electric power or military systems), we have not taken our design methodology to the point of

developing a complete and computationally feasible ECG analysis system, and indeed as we discuss in Section 4 a number of issues remain before such a system is developed. Rather what we have attempted to do is to use the ECG problem as a context in which we can explore the major problems in the design of distributed event estimation systems, present the elements needed in their solution and a systematic procedure for their application, and demonstrate the potential of this approach for solving a class of extremely complex estimation problems.

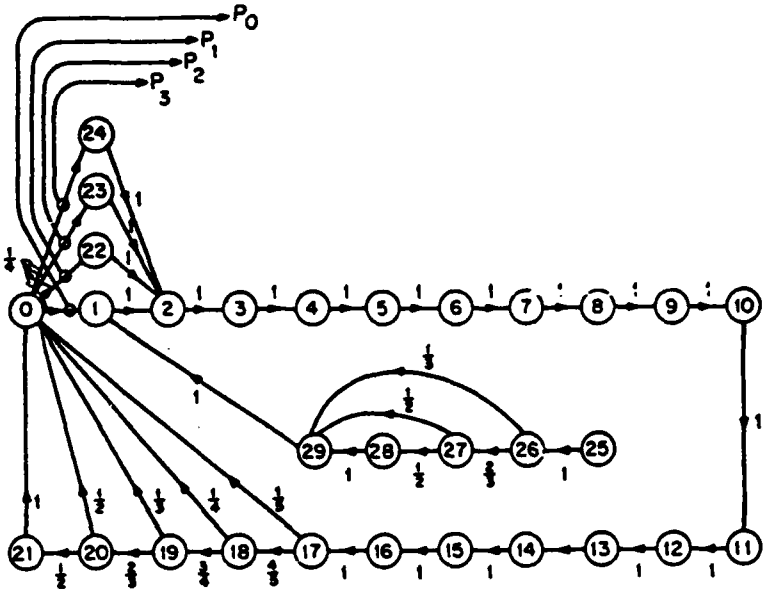
2. An Estimation Example

In this section we introduce our approach to estimation algorithm design by considering a particular example. The process whose state is to be estimated and which is illustrated in Figure 1 models normal cardiac rhythm with occasional reentrant-mechanism premature ventricular contractions (PVCs; these result from a normal excitation of the ventricles in effect circling back on itself and causing additional ventricular contractions). Let us briefly indicate several important features of the model:

(1) The model consists of two subprocesses one (the SA-atrial submodel, denoted C0, with state x_0) representing the behavior of the upper chambers of the heart and the other (the AV-ventricular submodel, denoted C1, with state x_1) capturing the behavior of the atrial-ventricular connection and the lower chambers of the heart. In the figure we have depicted the state transition probabilities for each submodel and the interaction between submodels which is captured by the dependence of these probabilities on the state of the other subprocess. We have also indicated which transitions in each subprocess give rise to waveforms or <u>signatures</u> that appear in the observed ECG. The signatures modeled are the P-wave (labeled P_0, P_1, P_2, P_3 in the

SA-Atria: state x_0

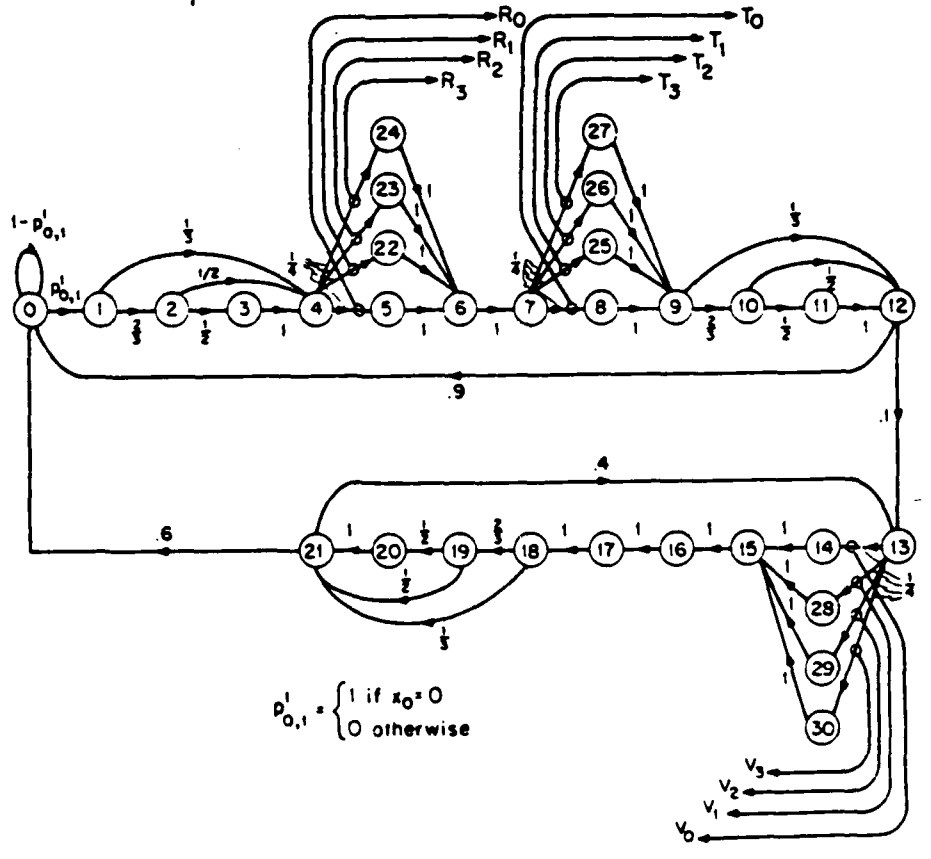
if $x_1 \neq 13$:



Call this tpm P^0

if $x_1 = 13$: then the tpm is $\frac{1}{2}(P^0+Q^0)$ where $Q^0 = \{q^0_{i,j}\}$ and $q^0_{i,j} = \begin{cases} 1 & j = 25 \\ 0 & \text{otherwise} \end{cases}$

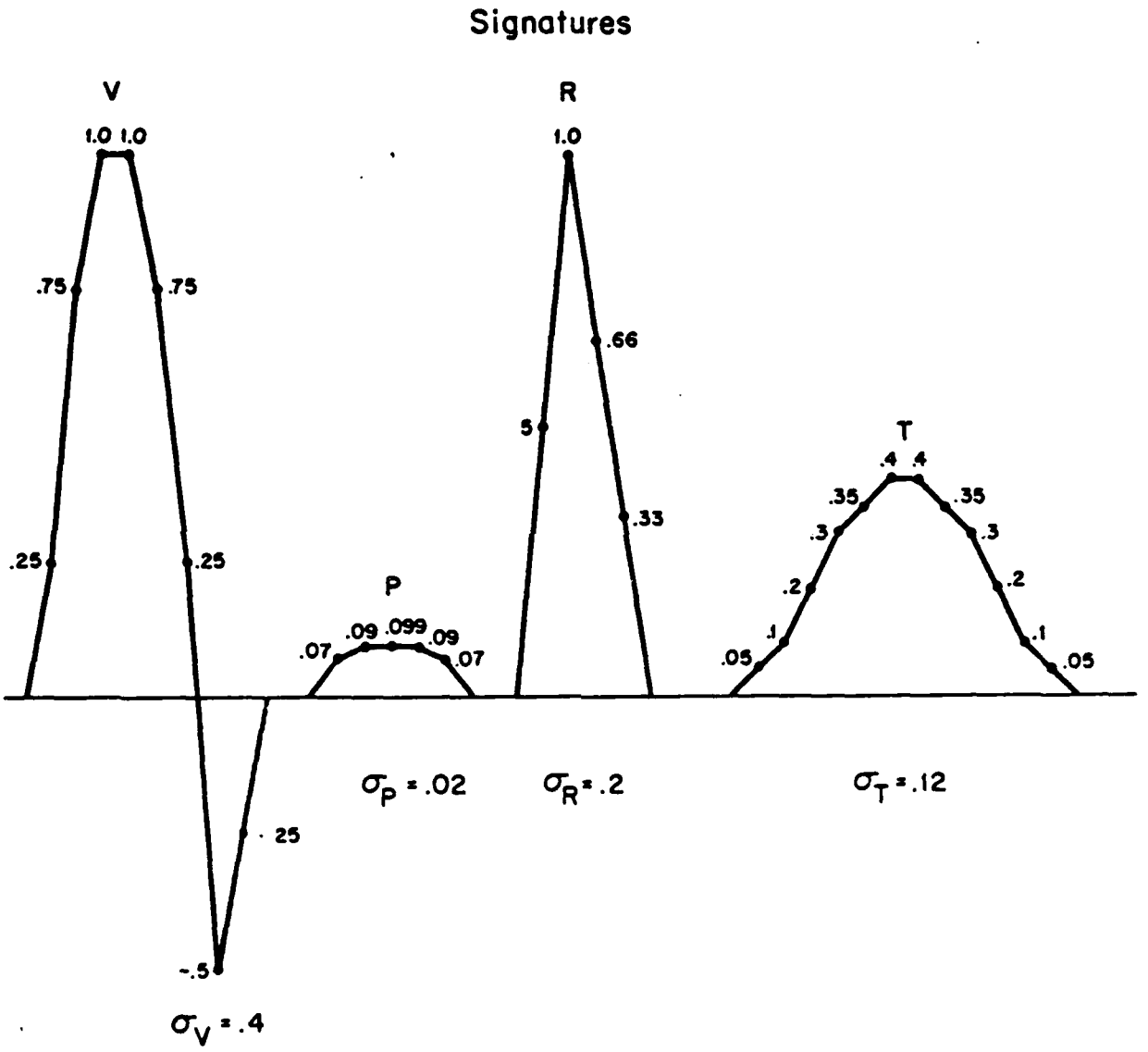
AV-Ventricles: state x_1



$p'_{0,1} = \begin{cases} 1 & \text{if } x_0 = 0 \\ 0 & \text{otherwise} \end{cases}$

(a)

Figure 1: A model of normal cardiac rhythm with occasional reentrant-mechanism PVC's: (a) the two subprocesses;



(b) the various signatures. Each occurrence of the P-, R-, T-, and V-waves consists of the signature plus zero-mean noise of standard deviations 0.02, 0.2, 0.12, and 0.4, respectively. In addition the entire ECG is observed in zero-mean noise of standard deviation 0.02.

figure), the R-wave, the T-wave (corresponding to ventricular repolarization following a normal contraction), and a V-wave corresponding to an aberrant reentrant PVC. The actual signature that appears in the ECG consists of a mean value plus additive white noise of specified variance. Each occurrence of these signatures includes a different realization of the noise. The means and variances for the various signatures are also illustrated in Figure 1.

(2) The interactions between the submodels are infrequent but are extremely strong. In particular, the diagram shown for the SA-atrial submodel represents normal activity which occurs unless $x_1 = 13$ in the AV-ventricular submodel, corresponding to the initiation of a PVC. When such an event occurs, it is possible for the electrical signal to propagate back to the upper chambers of the heart and in essence reset the timing of the heart's own pacemaker. This is captured by modifying the transition probabilities of x_0 so that with probability 1/2 x_0 is reset to state 25 when $x_1 = 13$, and with probability 1/2 x_0 proceeds in a normal fashion. In the x_1 submodel the only transition probability affected by the value of x_0 is p^1_{01}. In particular, $x_1 = 0$ represents the resting state of the ventricles, which is a trapping state ($p^1_{01} = 0$) until the ventricles are excited by an atrial contraction. This event is modeled by $x_0 = 0$ which both initiates the P-wave and excites the AV-ventricular submodel by causing p^1_{01} to change to a value of 1 for one time step.

(3) In our model, the ECG measurements are available at a rate four times the clock rate of the x_0, x_1 processes. In order to make it possible for signatures to start at any observation sample, each signature appears four times with 0, 1, 2, or 3 leading zeros in the mean and covariance sequences. (The subscripts on the wave-types indicate the number of leading zeros). This explains states 22-24 in submodel C0 and states 22-30 in submodel C1.

(4) The initiation of reentrant PVC's is modeled by transitions out of states 12 and 21 in submodel C1. Occupancy of state 12 corresponds to the completion of a normal R, T-wave pair, and from this state there is a probability of 0.9 of returning to the resting state and a probability of 0.1 of entering state 13 corresponding to the initiation of a reentrant PVC. Note that there is a much higher probability (0.4) of initiating

subsequent, consecutive reentrant PVC's (the 21-to-13 transition) which results in occasional occurrences of bursts of aberrant PVC's as are seen in episodes of ventricular tachycardia.

(5) The remaining states and transition probabilities model cardiac timing -- propagation delays, recovery time following contraction, etc. The model does allow for some uncertainty in this timing behavior and therefore some variability in the heart rate (which with a Markov chain cycle time of 0.04 seconds is, on the average, 75 beats per minute) as can be seen in the transitions with probabilities less than 1. It is certainly possible to add even more variability, but for simplicity we have not done that here.

Figure 2 shows a plot of several typical segments of a simulated ECG obtained using this model. Below the ECG tracing are several sets of annotations. The top row of annotations indicates the true times and types of waves that are present in the data (corresponding to the times at which transitions are made out of state 0 in submodel C0 (P-wave) and states 4 (R-wave), 7 (T-wave), and 13 (V-wave) of submodel C1). The remaining rows represent various annotations constructed during the estimation process, with the bottom row representing our final set of estimates. Before turning to a discussion of our estimation procedure, let us briefly comment on the nature of the simulated ECG itself. In particular, the simulated data resembles an ECG, but it is not entirely realistic. The chief difference is the character of the noise. First, no bandlimited noise, "shot noise" like events, or baseline instabilities are present. Second, the noise that is present, though it resembles electromyogram artifact when considered locally, appears to have a time-varying intensity (higher during the P-, R-, T-, and V-waves) which is not realistic for true electromyogram artifact. (As discussed in [Doerschuk 1986], the reason for the time-varying intensity was to give a crude model of the morphology variability of the different wave types). Each of these issues

Figure 2: Several segments of a simulated ECG obtained using the model in Figure 1. Annotations below the traces refer to estimates produced at several points in the estimation algorithm (see text).

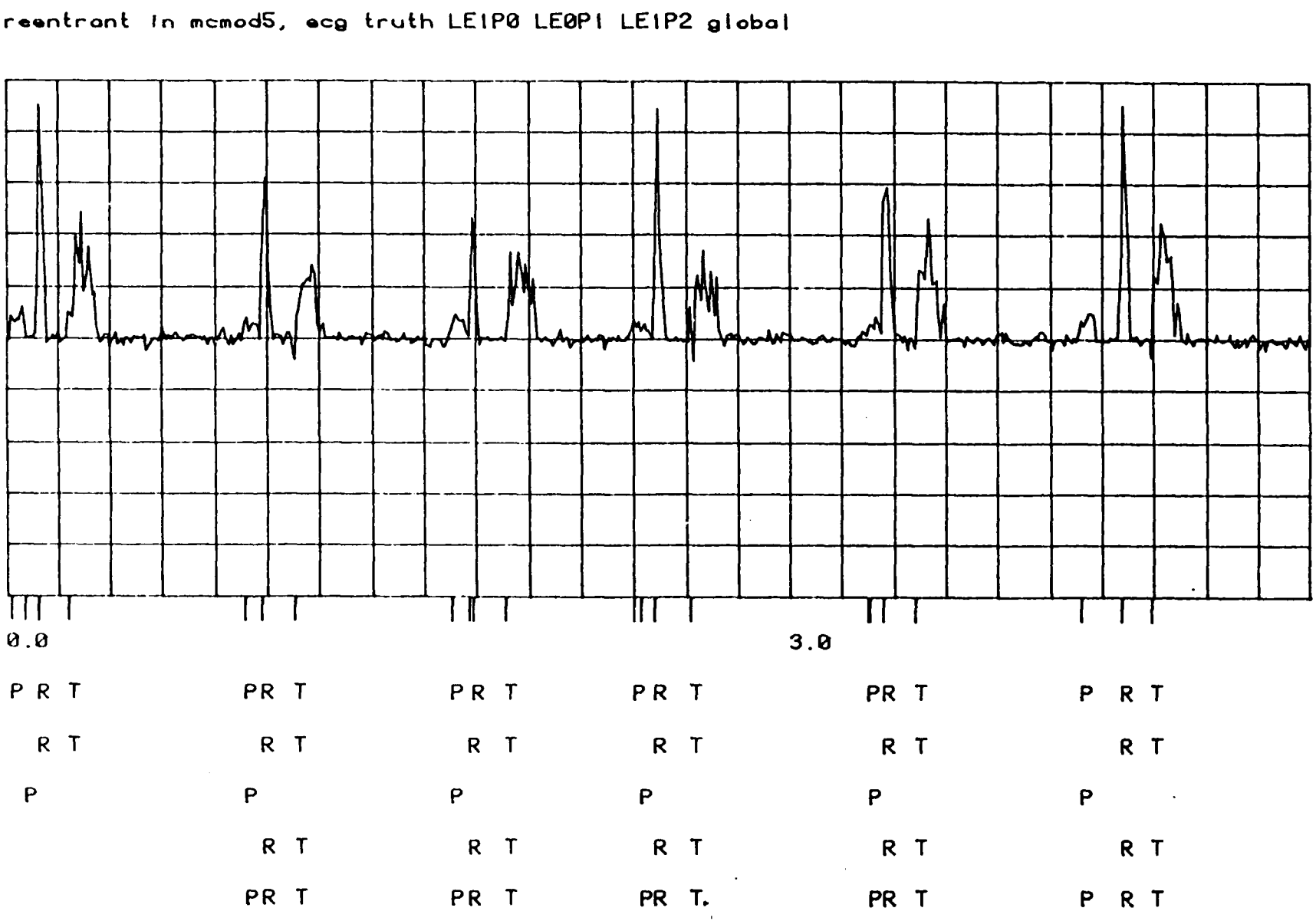


Figure 2: Several segments of a simulated ECG obtained using the model in Figure 1. Annotations below the traces refer to estimates produced at several points in the estimation algorithm (see text).

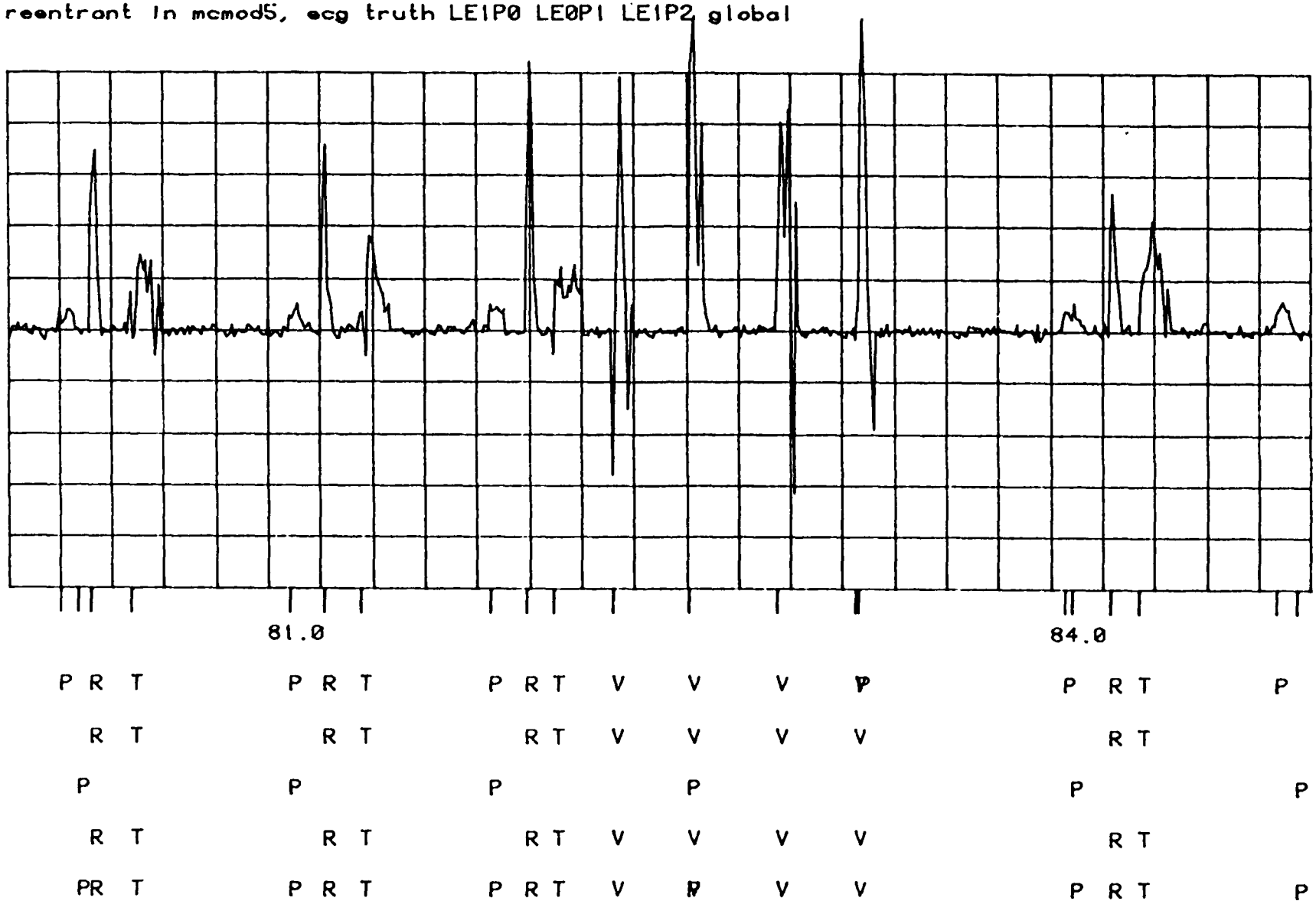


Figure 2: Several segments of a simulated ECG obtained using the model in Figure 1. Annotations below the traces refer to estimates produced at several points in the estimation algorithm (see text).

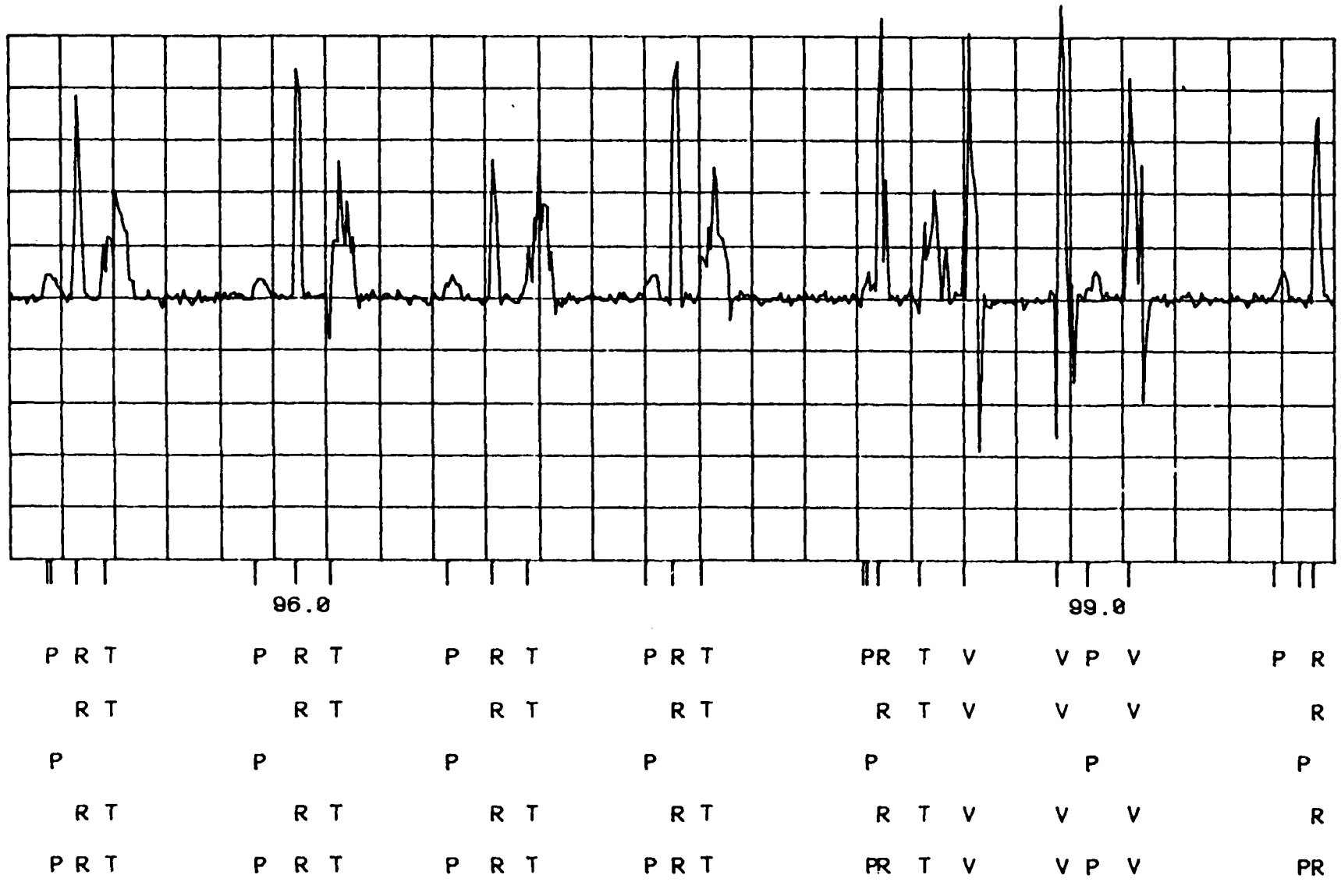
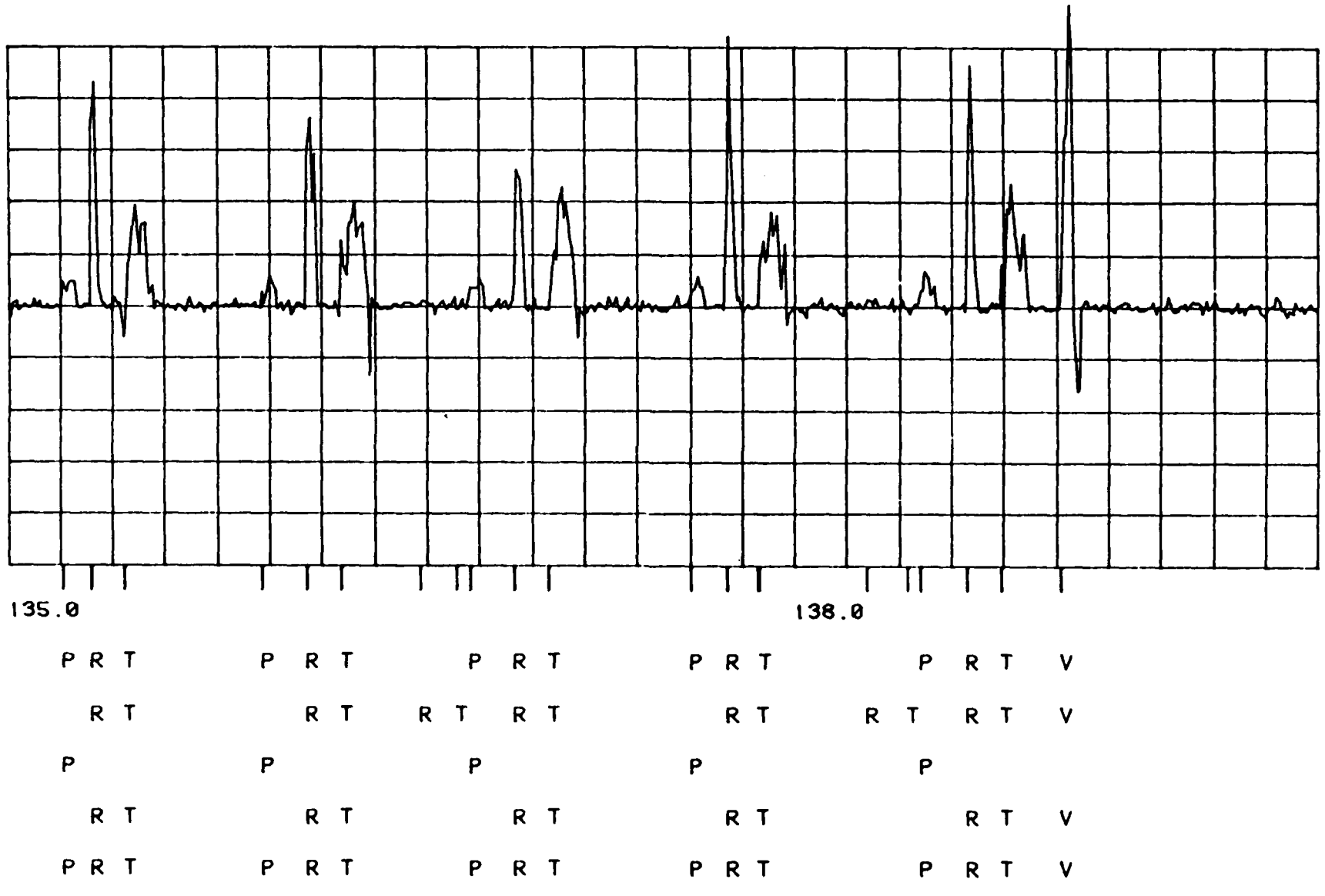


Figure 2: Several segments of a simulated ECG obtained using the model in Figure 1. Annotations below the traces refer to estimates produced at several points in the estimation algorithm (see text).



can be dealt with by adding additional detail to our model. Again for simplicity we have not done so here. Also, much of this detail may very well not be necessary for the purpose of designing rhythm processing algorithms.

Finally, in anticipation of the discussion to follow, we introduce some notation. Specifically, it is useful to have a compact pictorial description of interacting Markov chains. Such a description is illustrated in Figure 3. Here the label C0 denotes the SA-atrial submodel and C1 the AV-ventriclar submodel shown in Figure 1. The arrows between C0 and C1 indicate that the state of each subprocess influences the transition behavior of the other. Also, the arrows labeled P, R, T, and V indicate the waveforms initiated by each subprocess. In addition, we use the variables $h_{01}(n)$ to denote the sequence of interactions initiated by C0 and impinging on C1. That is $h_{01}(n)$ completely captures the influence C0 has on the transition probabilities of C1 for the transition $x_1(n) \rightarrow x_1(n+1)$. Referring to Figure 1, we see that we can define $h_{01}(n)$ so that it takes on only two values

$$h_{01}(n) = \begin{cases} 0 & \text{if } x_0(n) = 0 \\ 1 & \text{otherwise} \end{cases} \tag{1}$$

The only transition probability of C1 that is influenced by C0 is a trivial function of $h_{01}(n)$:

$$p_{01}^{1} = \begin{cases} 1, & h_{01}(n) = 0 \\ 0, & h_{01}(n) = 1 \end{cases} \tag{2}$$

Similarly we can define the interactions $h_{10}(n)$ from C1 impinging on C0

$$h_{10}(n) = \begin{cases} 0 & \text{if } x_1(n) \neq 13 \\ 1 & \text{otherwise} \end{cases} \tag{3}$$

so that if $h_{10}(n) = 0$ the transition probabilities are as indicated in the figure, and if $h_{10}(n) = 1$ they are the average of these values and a

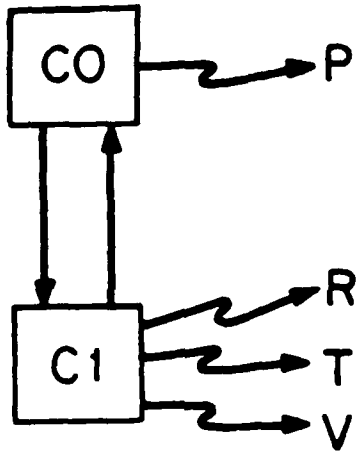


Figure 3: High-level block diagram representation of the model of Figure 1.

probability 1 reset to state 25 from any other state. Note that there are far fewer values for these interaction variables than for the corresponding states. This fact is used in an essential way in constructing several aggregate models used in our estimation methodology.

As discussed in the Introduction, our approach to state estimation for such a process involves the design of a set of interacting estimators, each of which focuses on estimation for a particular subprocess. Also, as we indicated, the existence of the interactions among subprocesses may require some iteration. For the present example our estimator can be viewed as consisting of three passes:

(1) Derive a preliminary estimate of ventricular activity (submodel C1).

(2) Based on the observed ECG and the estimates from pass 1, compute an estimate of atrial activity (submodel C0).

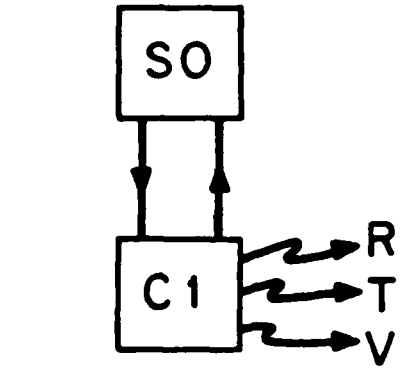
(3) Refine the ventricular estimate based on the observed ECG and the estimates of atrial activity from pass 2.

The results from (2) and (3) form the final estimate. Note that this approach roughly parallels the heuristic approach humans take in first identifying the high signal-to-noise ratio (SNR) events (R- and V-waves), then using these estimates to assist in locating the low SNR events (P-waves), and finally making adjustments to ensure accuracy and consistency. Also, while we describe these three steps as separate passes through the data, it is straightforward to construct a pipelined structure in which the three steps proceed at the same time.

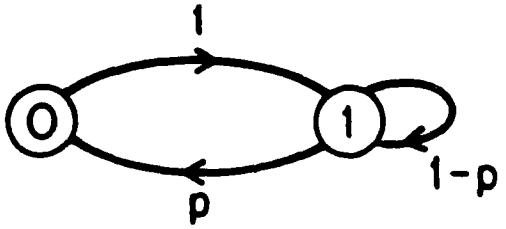
We now turn to a detailed examination of each of these three passes.

Because the first pass focuses on submodel C1, it is natural to include an exact copy of this submodel in the estimator's model. However, it is also necessary to model the interactions impinging on the C1 submodel, i.e. $h_{01}(n)$. Here one is confronted with a range of possibilities, from the exact model of CO depicted in Figure 1 (which provides complete accuracy in modeling interactions at the expense of computational complexity) to no model (which completely misses capturing interactions but leads to the simplest estimator). We have chosen in this study to use the simplest possible aggregate model for submodel CO with which we can still capture the full range of interactions with submodel C1. In particular the aggregate version of submodel CO that we use in this step is a two-state model, corresponding to the two possible values of $h_{01}(n)$. In addition, we allow submodel C1 to reset the state of our two-state aggregate model, again reflecting behavior seen in the full model. In the full discussion of our approach to estimation, this type of aggregate model is referred to as an "SO-submodel". Details for this example are given in Figure 4.

There are several further points to make about this first pass. Note first of all that the subprocess SO does not initiate P-waves. While it would certainly be possible to include the P-wave, this wave has a relatively small amplitude compared to the ventricular waves of primary concern in this pass. Consequently, there is little chance of confusing an unmodeled P-wave for an R- or V-wave, and for this reason we chose to ignore its presence in this pass. A second question that arises is the choice of the value of p in submodel SO. One can imagine several methods for choosing p -- matching some statistic of the exact submodel CO or viewing p as a design parameter to be



(a)



$$p = \begin{cases} .315789 & \text{if } x_1(n) \neq 13 \\ 0 & \text{otherwise} \end{cases}$$

(b)

Figure 4: Model for the first pass of the estimation algorithm; (a) overall block diagram; (b) detail of the SO model--state 0 corresponds to $h_{01}(n) = 0$, state 1 to $h_{01}(n) = 1$.

chosen to optimize estimator performance. In [Doerschuk 1985] several general statistical methods (which can be easily automated) are described for choosing parameters to match particularly useful statistics. In Section 3 we describe the statistical method used to obtain the value for p indicated in the figure. Finally, with this parameter specified we have a complete model, and the first step estimator is designed to produced a minimum probability-of-error state trajectory estimate for this model (i.e. estimates of the states of S0 and C1 as functions of time) based on the observed ECG. The method used to compute this sequence of estimates is the Viterbi algorithm [Forney 1973] which efficiently and recursively eliminates candidate trajectories that are suboptimal. Note that the estimate so-produced is an optimal smoothed trajectory, i.e. the best state estimate at each time is based on information before and after that time. The Viterbi algorithm minimizes the extent of the noncausality required in computing this optimal estimate.

The results of this first pass estimator are illustrated in the second row of annotations in Figure 2, where we have indicated the estimated times of occurrence of R-, T-, and V-waves. For the most part these estimates are quite accurate, thanks to the high SNR of these waves, although there are infrequent false alarms in the estimates caused by extra-long P-P intervals in which case the estimator attempted to match a T-wave with an actual P-wave.

The second step in our overall estimation structure is to estimate the state trajectory in the SA-atrial submodel. Therefore, it is natural to include an exact copy of the SA-atrial submodel in the estimator's model. For estimating this trajectory the only direct information from the ECG is the low SNR P-wave. However, there is also a great deal of indirect information

available through the causal relationship between P- and R- waves and V-and P-waves.

First consider interactions initiated by submodel C0. That is, consider the causality between P- and R-waves the latter of which only occur when the SA-atrial submodel successfully excites the AV-ventricular submodel. The question in this case is how we can exploit the auxiliary information concerning R-wave occurrences determined in the first estimation pass. At the very least one could imagine using the state estimates for S0 which are estimates of interactions impinging on the AV-ventricular submodel. Since the 0-state in this submodel corresponds to the 0-state in the original submodel C0 (and thus to attempts to excite submodel C1), the estimates of times at which S0 is in state 0 would be likely estimates of times at which $h_{01}(n) = 0$. However, because of the highly aggregated nature of S0, some of these estimates may be somewhat suspect. However, when such an estimate is coupled together with a closely following estimated occurrence of an R-wave (corresponding to the estimate of the C1 subprocess occupying state 4), the S0 estimate is much more likely to correspond to a true occurrence of an attempt at ventricular excitation. Consequently the information we provide to pass 2 from pass 1, which we will refer to as estimated <u>augmented interactions</u>, consists of the sequence of estimates of the states of S0 and C1 produced in pass 1.

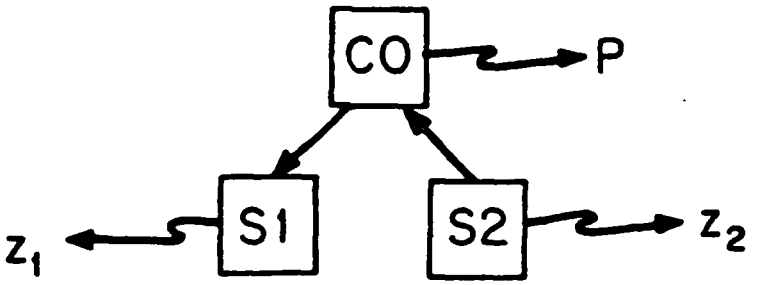
We now describe the issues involved in using these estimated augmented interactions. First of all, these estimates contain errors, so some uncertainty in them must be modeled. Note, however, that the errors of importance here are not only memoryless errors (which could be modeled by

static misclassification probabilities) but also errors in timing (e.g. the estimated time of occurrence of an R-wave may be in error by one or two samples). Consequently, we need a dynamic model for the way in which estimated augmented interactions provide information about C0. The way in which this is accomplished, as illustrated in Figure 5, is by modeling the estimated interactions, denoted by z_1, as the observed outputs of an additional submodel of a class we refer to as S1 submodels. This additional submodel receives interactions from submodel C0, whose state we wish to estimate. In order to model the fact that the estimates in $z_1(n)$ may contain time shifts relative to the actual values of the interactions $h_{01}(n)$, we take as the state of the S1 submodel a vector of the most recent interaction values. To minimize the size of the S1 state space one clearly wishes to minimize the dimension of this vector. For this study we found a dimension of 2 to be adequate, so that the state of S1 at time n is $(h_{01}(n-1), h_{01}(n-2))$. By examining the C0 submodel, we see that it is impossible for h_{01} to equal 0 at consecutive times. Thus, there are only three possible S1 states which we have coded as follows in Figure 5:

$$0 = (0, 1) \quad , \quad 1 = (1, 0) \quad , \quad 2 = (1, 1)$$

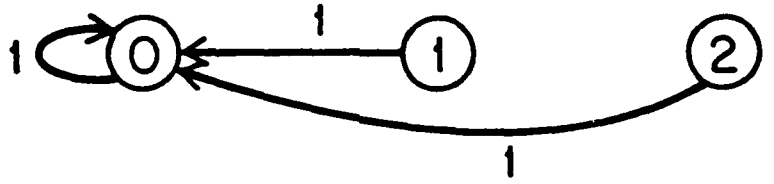
Since h_{01} is a deterministic function of the state of C0 it is straightforward to derive the way in which $x_0(n)$ affects the transition behavior of S1 (see Figure 5).

As in all of our models, the observation $z_1(n)$ is associated with transitions in the S1 subprocess which correspond to 3-tuples $(h_{01}(n-1), h_{01}(n-2), h_{01}(n-3))$ of interactions (corresponding to the states of the S1 subprocess before and after the transition). Our measurement model is then

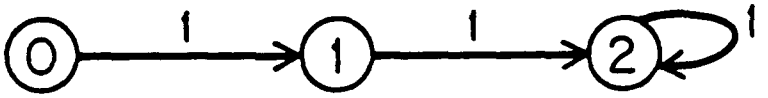


(a)

S1 Model if $x_0(n)=0$

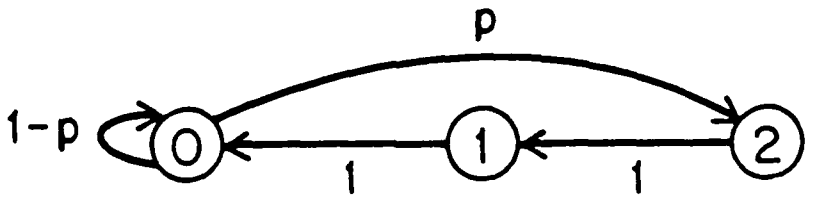


S1 Model if $x_0(n) \neq 0$



(b)

S2 Model (p=0.00784)



(c)

Figure 5: Model for the second pass (the submodel C0 is reset, i.e, its transition rates are as given in Figure 1 with $x_1 = 13$, only if the S2 process is in state 2.)

the set of conditional probabilities

$$\Pr(z_1(n) | h_{01}(n-1), h_{01}(n-2), h_{01}(n-3)) \tag{4}$$

Since the Viterbi algorithm provides us with noncausal estimates, we are free to build some noncausality into this model. Consequently, we have chosen to take $z_1(n)$ as the pass 1 estimate at time n-2, which therefore provides an estimate of $h_{01}(n-2)$. Thus the model allows us to capture time shifts of ± 1.[1] The specification of (4) can be obtained by analysis of the performance of the first step estimator. We have estimated these quantities by simulating that estimator and tabulating the results.

We now must consider the interactions $h_{10}(n)$ initiated by C1 and impinging on C0, i.e. the effect of V-wave occurrences on C0. There is a similarity here with the modeling of S0 in the first pass but in the present context we also have the estimates from pass 1 which tell us something about these interactions. Specifically, since we used the exact C1 submodel in pass 1, we can deduce estimates of h_{10} (see (3)). We take these estimates as our observation z_2 for pass 2 (without any augmentation as was done for z_1 since the first step estimator used an exact model for C1 and consequently should produce comparatively accurate estimates). Also, as with the S1 submodel, we need to model possible estimation timing errors, so again we take the state of S2 to be a set of the most recent interactions, in this case $(h_{10}(n), h_{10}(n-1))$.[2] In this example it is impossible for h_{10} to equal 1 at two

---

[1] Note that allowing timing errors of both signs is needed in any realistic algorithm since, for example, wave detection preprocessors are noncausal in nature.

[2] Note that there is some asymmetry in comparison with the S1 submodel where the state was lagged one step. This is a result of the fact that in the S1

consecutive times, and thus we can code the feasible S2 states as

$$0 = (0, 0) \quad , \; 1 = (0, 1) \quad , \; 2 = (1, 0)$$

In this example the C0 submodel transition probabilities are as illustrated pictorially in Figure 1 for $x_{S2}(n) = 0$ or 1 and incorporate the 0.5 probability reset to state 25 when $x_{S2}(n) = 2$. The model for S2 is also illustrated in Figure 5. Note that as with S1, there is a parameter p to be chosen to specify the S2 transition probabilities. This parameter was also chosen to match statistics of the true h_{10} process (as were the other probabilities of the S2 process, but these can be determined trivially from the structure of the C1 submodel). The general method used is described in the next section. Finally, the observation $z_2(n)$, which is the pass 1 estimate of $h_{10}(n-1)$, is modeled as resulting from S2-transitions. Thus again we must specify a distribution, namely

$$\Pr(z_2(n) | h_{10}(n), \; h_{10}(n-1), \; h_{10}(n-2))$$

which we have again done by simulation.

This completes the specification of the second pass model. It is worth noting, however, that there are several things that have been left out of the model. The most glaring of these, perhaps, is the complete absence of R-, T-, and V-waves. For the pass 1 estimation algorithm we argued that it was reasonable to consider omitting P-waves from the model since (a) we were focusing most attention on submodel C1 and (b) the P-waves were of low amplitude. In pass 2, the first argument holds (here we are focusing on C0),

---

submodel, $h_{01}(n)$ is a deterministic function of $x_0(n)$. Thus for the state $x_0(n)$ to correctly "influence" the next transition in S1, we needed to introduce the time delay in defining the S1 state. This is not needed in S2, since there is no such deterministic coupling.

but the latter does not, since R-, T-, and V-waves are not of low amplitude. In the general procedure described in the next section, we allow for the possibility of taking such waves into account through so-called subtractor submodels. However, as the results in this section and in [Doerschuk 1985] indicate, for ECG-type models such as the one considered here the use of such models is of doubtful value. Intuitively the reason we can ignore such waves in the pass 2 estimation algorithm is that through $z_1(n)$ and $z_2(n)$ we are providing indications of the times at which these waves occur. Given then the coupling between these waves and the likely times of P-waves, captured in the original C0-C1 model and in our simplified pass 2 C0-S1-S2 model, the pass 2 estimator will not try to account for R-, T-, and V-waves by placing P-waves in their locations.

A second issue we have ignored is that of allowing the C0 submodel to influence the S2 submodel (which is, in our approach, autonomous). Since the C0 submodel does influence the C1 submodel, there would seem to be some argument for doing so. However, it is precisely this influence that is focused upon in the S1 submodel, while the S2 submodel focuses on that part of the C1 submodel, dealing with V-waves, which is unaffected by the C0 submodel. Consequently, while our general modeling methodology allows C0 to influence S2, it is not necessary to include this bit of complexity in the present context.

Finally, we note that in our model we are considering z_1 and z_2 to be independent measurements, which is clearly erroneous since they are both determined by the pass 1 estimation process. One can certainly construct a somewhat more complex model involving a joint distribution of z_1, z_2 given the

combined information in the most recent transitions of S1 and S2, but this was not found to be necessary here (since again z_1 and z_2 focus on different portions of the overall C0-C1 model).

In summary, the second pass of our estimation procedure consists of the minimum probability-of-error estimation of the state trajectory of the model given in Figure 5 (using the Viterbi algorithm) given the ECG measurement and the derived measurements z_1 and z_2 from the first pass. The results for this example are given in the third row of annotations in Figure 2 where we have indicated the times at which P-waves were estimated to have occurred. Comparing this to the top row of annotations we see that performance is quite good. Note that the erroneous R,T-wave pairs from pass 1 near 136.6 and 138.3 seconds did not lead to an erroneous P-waves in pass 2, thanks to our modeling of z_1 which incorporated the possibility of such false alarms. Note also the occurrence of P-wave timing errors (as illustrated near 80.2 and 99.9 seconds) all of which underestimate the P-R interval. Finally, note that it is possible in our model (and in the heart) for P- and V-waves to occur nearly simultaneously or for V-waves to preempt an already occurring P-wave from initiating a normal R-wave. Having knowledge of this, the pass 2 estimator will attempt to insert P-waves when the timing seems likely even though the presence of V-waves may obscure the P-wave (indeed this is precisely how a human would create such estimates). An example of correct estimates of this type can be found near 99 seconds. A false alarm can be seen near 82.6 seconds, and a missed detection near 83.3 seconds. While the value of such estimates is suspect (and not of particular consequence) they do provide rather graphic examples of the use our estimator makes of the timing and

control information embedded in our models.

The third pass of the estimation process, whose purpose is to provide improved and consistent estimates of ventricular activity, is based on a model, illustrated in Figure 6, with structure analogous to that of pass 2 (with the roles of submodels C0 and C1 interchanged). Specifically, since the focus of pass 3 is on submodel C1, we include a complete version of this submodel. Also, from pass 2 we obtain estimates of interactions h_{10} initiated by C1 and impinging on C0 (these come from the estimates of the state of the S2 submodel in pass 2) and estimates of interactions h_{01} initiated by C0 and impinging on C1 (these come directly from the pass 2 estimate of the state of C0). As before, since only an aggregate model (S2) is used in pass 2 to estimate h_{10}, we augment these estimates with the corresponding estimate of the state of C0. These augmented estimates are incorporated in pass 3 as observations, denoted by z_1 in Figure 6, of an S1 submodel whose state is the most recent two values of the actual interactions, $(h_{10}(n-1), h_{10}(n-2))$. The pass 2 estimates of h_{01} are incorporated in pass 3 as the observations, z_2, of an S2 submodel whose state is $(h_{01}(n), h_{01}(n-1))$. The same methods as in pass 2 were used to compute the transition probabilities and measurement distributions. The estimator is again a minimum probability-of-error estimator using the ECG and the derived measurements z_1, z_2.

The result of applying this estimator is illustrated in the fourth row of annotations in Figure 2. The final, overall estimate consists of the C0-state estimate of pass 2 (row 3) and the C1-state estimate of pass 3 (row 4), which are combined in row 5. Comparing the top and bottom rows we see that the estimator has performed quite well. Disregarding the initial heartbeat (which

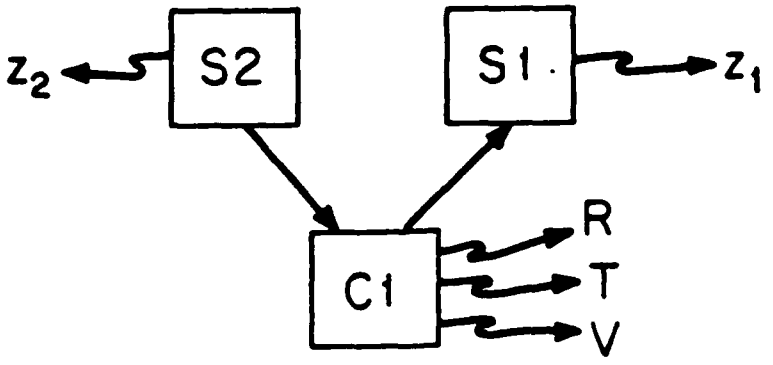


Figure 6: Block diagram of the model for the third pass.

was missed in pass 3 because of the specific way in which we implemented the initialization of the latter passes of our algorithm) all R-, T-, and V-waves were detected and located with no false alarms. Note that while there had been several false R, T-wave estimates in pass 1, these have been completely eliminated in pass 3, in which we have the benefit of using estimates of CO-behavior in order to enforce consistent overall estimation.

The estimation of P-wave occurrences is also quite good. Quantifying this performance, however, is an interesting question itself, since one is clearly not just interested in estimation errors at points in time but also in timing errors at points in the estimated event cycle -- i.e. an estimation error of one time sample in locating a P-wave should not be thought of as a missed detection but rather as a (not particularly troubling) timing error. Much more on the issue of performance measures for event-oriented estimation problems can be found in [Doerschuk 1985]. This example does, however, indicate the main ideas. In examining the results of the full simulation we find that there are only two isolated false positive P-wave indications and one isolated false negative (neglecting the initial heartbeat), where by "isolated" we mean that there is no nearby P-wave in the true or estimated state trajectories. Given that there are 230 heartbeats in this simulation, these correspond to a false positive rate of .009 and a false negative rate of .004. There are also 23 other paired false positives and negatives, where we have used the criterion of associating estimated and actual P-wave locations only if the waveforms at these locations overlap. This corresponds to a paired error rate of 0.10. Note that in our model, every R-wave must be preceded by a P-wave, and thus this pairing is to be expected (we discuss

a less deterministically-coupled model in [Doerschuk 1985]). It is worth noting that in each of these paired errors, the estimated P-wave location was closer to the R-wave than the true R-wave, indicating a bias that may be removable (and is most likely due to the pass 2 estimator correlating the P-wave with the initial portion of the R-wave).

In [Doerschuk 1985] we consider a variety of other models with other types of variability than that captured in the model of Figure 1. For example, we have examined models with transient AV block, i.e. models in which not every attempt at ventricular excitation leads to an R-wave even if the ventricles are apparently in the resting state. Because of the additional freedom in the model, one would expect some drop in performance. However the drop is extremely small both for a globally optimal estimator and for the suboptimal estimator designed based on the principles outlined in this section and formalized in the next.

3. A General Design Methodology

The example described in detail in the previous section illustrates the major elements of a general estimator design methodology for distributed Markov chains. In this section we describe that methodology. Specifically, consider the estimation of an interconnection of subprocesses, which we denote $C_0, C_1, \ldots, C_N$, with states $x_0, x_1 \ldots, x_N$, given measurements of signals containing signatures corresponding to particular state transitions in these subprocesses. The interactions among these subprocesses are defined as follows. Let $h_{ij}(n)$ denote the interaction initiated by C_i and impinging on C_j at time n. This interaction is a deterministic function of $x_i(n)$, and the

transition probabilities of C_j are deterministic functions of $\{h_{ij}(n) | i \neq j\}$. The assumption is that the set of possible transition probabilities for each C_j (and thus the set of possible values of $\{h_{ij}(n) | i \neq j\}$) is quite small.

Our overall estimator consists of an interconnection of local estimators (LE's), each of which focuses on the estimation of one of the subprocesses. Because of the existence of interactions with and events in the observed data due to other subprocesses, each LE not only must take these effects into account in its model but also must communicate with the other LE's in order to determine a consistent and accurate set of estimates. Consider first the initial pass through the data, at which time the LE's have no previous information to communicate. Let us focus on the LE for a specific submodel, namely C_j. In order for this LE to construct its initial estimate it in general it will need:

(a) A complete model of the subprocess C_j on which it is focused.

(b) A model of the interactions impinging on C_j.

(c) A model of the waveforms generated by the other submodels.

The model referred to in (b) is called an SO submodel, and a major objective is to make it as simple as possible in order to keep the LE as simple as possible.[3] In the example in the previous section and throughout

---

[3]There are two distinct ways in which one can perform this modeling step and several that follow. In particular, in this section we describe the construction of a single SO submodel capturing the interactions impinging on C_j from all other subporcesses. In [Doerschuk 1985] an analogous approach is described for constructing separate SO submodels for the interactions initiated by each of the other subprocesses.

our work we have taken the states of the SO submodel to be in one-to-one correspondence with the possible values of the N-tuple $\{h_{ij}(n) | i=j\}$. Whether this or a more complex model is used to describe the dynamics of these interactions, the question remains of choosing the transition probabilities for the SO submodel. Our approach has for the most part been to match these one-step transition probabilities to the actual steady-state versions within the original process, that is, to

$$\lim_{n\to\infty} \Pr[\{h_{ij}(n) | i\neq j\} | \{h_{ij}(n-1) | i\neq j\}, \{h_{ji}(n-1) = h_{ji} | i\neq j\}] \qquad (5)$$

This computation deserves some comment. Note first that we have included conditioning on $\{h_{ji}(n-1) | i\neq j\}$, which reflects the influence C_j has on the other subprocesses. This results in the transition probabilities of SO being influenced by the state of C_j. Again we typically expect this influence to manifest itself as a small number of possible values for a small subset of the transition probabilies (e.g. in our case study only the parameter p in Figure 4 is influenced, and it only takes on two values). Note that conditioned on $\{h_{ji}(n) | i\neq j\}$, $\{x_i(n) | i\neq j\}$ is a Markov chain. However this is not typically the case for the highly aggregated $\{h_{ij}(n) | i\neq j\}$, so that the limit in (5) is not a completely trivial computation. On the other hand, once we compute the ergodic probabilities for $\{x_i(n) | i\neq j\}$, it is straightforward to compute (5).

Typically for models with infrequent changes in interactions, most of the transition probabilities specified in (5) are 0 or 1, and there are only a few parameters (such as p in Figure 4) for which this computation is necessary.[4]

---

[4]Indeed for all of the cases considered in [Doerschuk 1985], the model was exactly as in Figure 4 (with different values of p), since in all of our cases

Finally, we note that there are some cases in which the matching of the steady-state statistic (5) may be inappropriate. In particular, in using (5) we are in essence assuming that the transition probabilities of $\{x_i(n) | i \neq j\}$ do not change very frequently (so that steady-state is actually achieved) -- i.e. that the time variations observed in the actual $x_j(n)$ process do not lead to frequent changes in the interactions $h_{ji}(n)$ (which are assumed to be constant in (5)). We refer the reader to [Doerschuk 1985] for examples violating this assumption and in which we must set the SO transition probabilities in a different manner. Note that this assumption is in fact violated in our case study. In particular, while it is certainly true that $h_{10} = 0$ for long periods of time, since this corresponds to x_1 being <u>any</u> state other than 13, $h_{10} = 1$, corresponding to $x_1 = 13$, cannot possibly occur at any <u>two</u> consecutive times. In this case, since $h_{10} = 1$ corresponds to a reset of CO to state 25, and since all states in CO other than 0 correspond to $h_{01} = 1$, it is reasonable to reset the state of SO to 1 whenever $x_1 = 13$. This is what is specified in Figure 1 <u>and</u> what we would calculate from (5). Thus (5) is often useful even if the assumption on which it is based is violated.

Let us now discuss the model referred to in (c). This is one of the <u>subtractor submodels</u>, denoted by S3, referred to in the previous section, which is incorporated in order to keep the LE from interpreting waveforms generated by other submodels as coming from C_j. In essence what we would like to do is to present the LE with observations containing only those signatures generated by C_j. Since this is not possible, we equip the LE with a mechanism

---

there have been only two interaction values, one of which could not occur at consecutive times.

for estimating when other signatures have occured so that it can in effect subtract out their effects. In general, one can construct a separate S3 submodel for each signature not initiated by C_j. While it would certainly be possible to couple these subprocesses with the C_j and S0 submodels, we have obtained good results with a simpler structure in which each S3 submodel is a completely autonomous, aggregated process that produces interarrival statistics for the wave of interest identical to those produced by the exact model. More precisely, we have always taken S3 submodel to be of the form shown in Figure 7. The two parameters are chosen as follows. Let $\tau_{SS}(n)$ denote the time between the nth and (n+1)st occurrence of the signature S in the original process. Then we choose p and q to match the probability that signatures occur at successive times and the mean time between successive signatures. That is

$$p = 1 - \lim_{n\to\infty} \Pr[\tau_{SS}(n) = 1] \tag{6}$$

and

$$\frac{p}{q} + 1 = \lim_{n\to\infty} E[\tau_{SS}(n)] \tag{7}$$

Again the statistics in (6),(7) can be calculated from the ergodic probabilities of the full model. In most cases $\Pr[\tau_{SS}(n) = 1] = 0$, so that

$$q = \frac{1}{\lim_{n\to\infty} E[\tau_{SS}(n)] - 1} \tag{8}$$

Therefore, in our general design methodology we construct each initial LE model using C_j, S0, and S3 components as illustrated in Figure 8 and compute the initial pass minimum probability-of-error estimates for each LE. We are then in a position to consider a refinement pass, in which each LE reprocesses the data, together with information provided from the initial passes of the LE's.

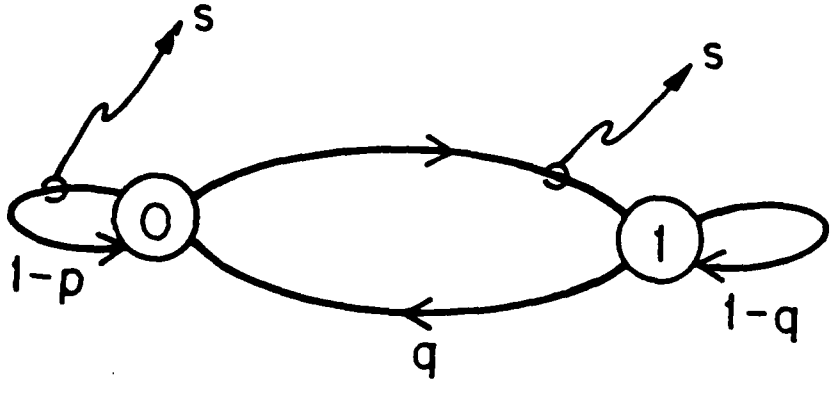


Figure 7: An S3 chain. Here the 0 - 0 and 0 - 1 transitions initiate the signature denoted by S.

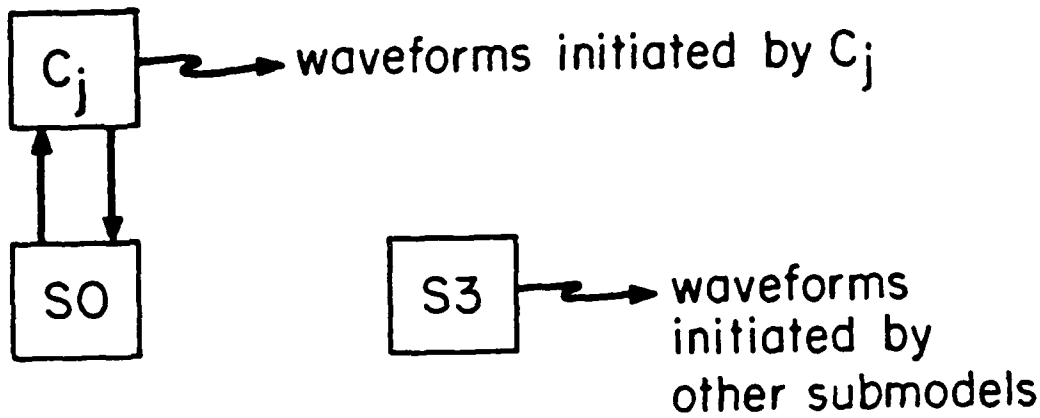


Figure 8: The structure of a general LE model for an initial pass.

The LE for subprocess C_j will in general need the following elements in its model for a refinement pass:

(1) A complete model of C_j.

(2) A model of the information provided by the previous pass concerning interactions initiated by C_j.

(3) A model of the information provided by the previous pass concerning interactions impinging on C_j.

(4) A model of the information provided by the previous pass concerning times of occurrence of waveforms generated by the other subprocesses.

Note that (2) and (3) together corresponds to (b) in the initial pass. We have chosen to split these two models here as (i) it makes the modeling of the information simpler and (ii) the information referred to in (2) and (3) may typically come from different sources or be of very different accuracy or structure (given that each LE has an accurate model of its own subprocess but only highly aggregated models of the others).

As discussed in the previous section, the models referred to in (2) and (3), which we refer to as S1 and S2 submodels, respectively, must capture the timing and estimation uncertainties from the previous pass. Each accomplishes this by taking as its state space a moving window of the most recent interactions. In particular, the state of the S1 submodel consists of a window of the most recent values of the N-tuple

$$\{h_{ji} \mid i \neq j\}$$

while the state of the S2 submodel is a window of the most recent values of

the N-tuple[5]

$$\{h_{ij} | i \neq j\}$$

An objective in designing these models is to keep the window lengths K_1 and K_2 small in order to minimize state space size. This desire is balanced by the need to model estimation timing errors (since the maximum such symmetric error that can be modeled corresponds to 1/2 the window length). In our work we have always taken this window length equal to 2.

Consider next the dynamics of the S1 and S2 submodels, both of which are specified in a straightforward manner. In particular, since each $h_{ji}(n)$ is a deterministic function of $x_j(n)$ and since the full C_j model is used by the LE, the S1 dynamics are essentially a shift register memory. That is, given $x_j(n)$, the transition

$$x_{S1}(n) = \{h_{ji}(m) | i \neq j,\ m=n-K_2,\ldots,n-1\}$$

$$\downarrow$$

$$x_{S1}(n+1) = \{h_{ji}(m) | i \neq j,\ m=n-K_2+1,\ldots,n\}$$

is deterministic (i.e. for each present state there is one next state [whose identity depends on $x_j(n)$] that S1 will occupy with probability 1).

The dynamics of the S2 submodel are not quite this trivial. Specifically, as in the S0 submodel, we choose the transition probabilities to match those in the original process. In particular, in analogy with (5), we can choose these to equal

$$\lim_{n\to\infty} \Pr[\{h_{ij}(m) | i \neq j,\ m=n-K_2+2,\ldots,n+1\} | \{h_{ij}(m) | i \neq j, m=n-K_2+1,\ldots,n\}, \{h_{ji}(n) | i \neq j\}] \tag{9}$$

---

[5]Recall from the previous section that there is some asymmetry in the windows here, with the window for S1 stopping at time n-1, and the window for S2 stopping at time n.

By including the conditioning on $\{h_{ji}(n)|i\neq j\}$ we can capture the interactions initiated by C_j and impinging on the other subprocesses (and therefore, in the LE model, on S2). However, as discussed in the previous section, the effects of these interactions are the primary concern of the S1 submodel, and thus it is worth seeking and typically possible to find a far simpler model. In fact throughout our work we have been able to completely eliminate the influence of C_j on S2 (which then operates autonomously, generating the interactions that impinge on C_j). This can be done by using (9) with $\{h_{ji}(n)|i\neq j\}$ set equal to the values that represent the most usual interaction.[6] Another approach is to compute the average of (9) over the possible values of $\{h_{ji}(n)|i\neq j\}$ using their ergodic probabilities. In our work we have used the latter of these two methods.

Consider next the modeling of the "measurements" provided by the previous data pass. With respect to S1, we have in general the following sources of information concerning the interactions initiated by C_j:

(i) The previous state estimate of C_j from its associated LE. From this we can directly compute an estimate of $\{h_{ji}(n)|i\neq j\}$.

(ii) The augmented interactions from each of the other LE's. These consist of the estimate of the interaction impinging on the C_i submodel associated with each LE (obtained from the aggregated S0 submodel used by the LE) and the corresponding C_i-state estimate (see the previous section for a discussion of why this augmenting information is included).

---

[6] In our ECG examples this corresponds to no attempt at interprocess excitation, as such electrical excitations occur over relatively short time periods (usually a single time sample).

Together this information forms a measurement, which we denote $z_1(n)$, and, since we associate measurements with transitions, we model the information contained in $z_1(n)$ by

$$\Pr[z_1(n) | \{h_{ji}(m) i \neq j,\ m=n-K_2-1,\ldots,n-1\}] \tag{10}$$

As discussed in the previous section, since we have introduced some noncausality with the Viterbi algorithm, we have the flexibility of doing so here (and the need in order to model positive and negative timing errors). Consequently, we take $z_1(n)$ to be the previous pass estimates indicated in (i) and (ii) evaluated at time $n-1-K_2/2$. Finally, while it is possible to devise analytical methods to obtain approximations for (10), we have found it easier to evaluate these distributions by simulation.

For S2, we have the following sources of information concerning interactions impinging on C_j:

(i) The augmented estimated interaction provided by the previous pass of the LE for C_j.

(ii) The estimated state of each C_i provided by the associated LE. From these we can directly compute estimates of each $h_{ij}(n)$.

This information forms the measurement $z_2(n)$, which is modeled via

$$\Pr[z_2(n) | \{h_{ij}(n) | i \neq j,\ i=n-K_2,\ldots,n\}] \tag{11}$$

As in the case of S1, we introduce some noncausality by taking $z_2(n)$ to be the previous pass information evaluated at $n - K_2/2$, and we determine (11) by simulation.

Finally, consider modeling the information available from the previous pass concerning waveforms generated by other submodels. Each such waveform is

modeled by what we call an S4 submodel, which is a second class of subtractor models used in refinement passes. S4 submodels resemble S2 submodels in structure and principle. Consider an S4 submodel corresponding to a particular waveform generated by submodel C_i. The measurement $z_4(n)$ provided by the previous pass LE for C_i is a sequence of binary annotations -- 0 if the LE estimates that the particular C_i waveform was not initiated at that time sample and 1 if the estimate is that the waveform was generated. The state of the S4 submodel is a window of the most recent true values of these binary annotations. As with S2, the transition rates of this model are chosen to match the corresponding transition rates of sequences of binary annotations in the full model. If the counterpart to (9) is used, the S4 model will in general be influenced by C_j. Again as in the case of S2, we have typically simplified this model so that S4 is autonomous, by averaging out the C_j-dependence using the ergodic distribution for $x_j(n)$.

The output of the S4 chain is a sequence of occurrences of the waveform being modeled. Such outputs occur at all S4 transitions to states with a 1 as the most recent annotation. The auxiliary observation $z_4(n)$ is again modeled via a probability distribution conditioned on the most recent S4 transition. We have determined distributions of this type via simulation.

The structure of the models on which each LE refinement pass is based is depicted in Figure 9. Again the Viterbi algorithm is used. In general one can envision making several refinement passes, with the final estimate consisting of the collection of C_j-state estimates from the final passes of the corresponding LE's. The primary purpose of the refinement passes is to improve the accuracy and consistency of this set of estimates. In particular,

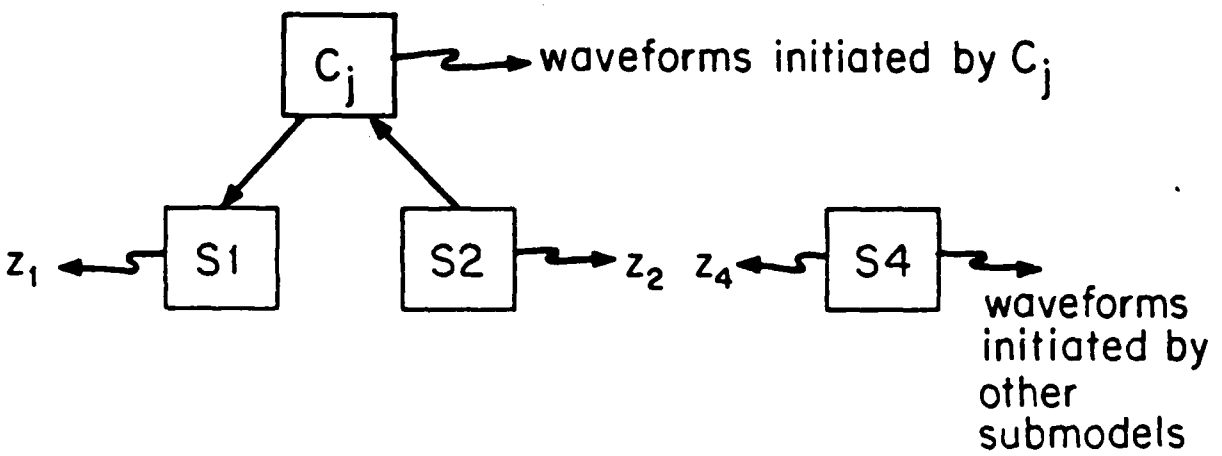


Figure 9: The structure of a general LE model for a refinement pass.

if one implemented a single, optimal estimator for the full process, one would know for certain that all transitions present in the final state estimate would be consistent (i.e. have nonzero probability in the full process). When one uses a collection of distributed, simpler LE's, there is no such guarantee, but the coordination made possbile by refinement passes makes the occurrence of inconsistent estimates far less likely. In fact, in our studies consistency has not proven to be a problem.

The complete procedure we have described requires the implementation of a full set of LE's for the initial pass (based on models as in Figure 8) and subsequent refinement passes (each based on its own model as in Figure 9). As in our example in Section 2, it is typically possible to simplify this design considerably. First of all, as in our example, for each LE it is often not necessary in the initial pass to include subtractor submodels S3 for waveforms of low SNR compared to the waveforms generated by the submodel corresponding to the LE. Also, as we showed, it may not be necessary to include <u>any</u> S4 submodels, since the information provided through S1 and S2 submodels essentially provides timing information that allows the LE to avoid intervals in which these interfering signatures may appear. Doerschuk [1985] presents comparative results with and without S3 and S4 submodels that support these simplifications.

It is also typically possible to eliminate many of the LE's from each pass. For example, in the initial pass, one typically would implement LE's only for submodels generating the higher SNR signatures (such as R-waves), as the performance of initial pass LE's for other submodels with only low SNR signatures (or <u>no</u> signatures, as is the case for some rhythm models described

in [Doerschuk 1985, 1986]) will generally be unsatisfactory. Also, in order to achieve consistency, we do not need to refine all LE's in subsequent passes. In particular we typically can implement an alternating iterative structure much as in the example in which we initially estimate the C_j with high SNR signatures, then use these estimates to assist in estimating only the remaining C_i during the next pass; these estimates can then be used in turn during the following pass in the re-estimation of the C_j from the initial pass in order to improve the accuracy and consistency of the C_j-state estimates. Note that in addition to eliminating entire passes of LE's, such a structure reduces the quantity of z_1 and z_2 measurements to be processed by the remaining LE's. In fact, the full set of such information described previously has some redundancy, reflecting the fact that perhaps not all of this intermediate processing is needed. The structure described above simplifies the design by removing these redundant sources of information. Doerschuk (1985) presents results favorably comparing reduced designs of this type to estimators incorporating more or all of the LE's at each stage.

4. Conclusions

In this paper we have presented a methodology for the distributed estimation of interconnected finite-state processes given the observation of signals containing waveforms initiated by events in the various processes. Our motivation for examining this class of models is the problem of automated ECG analysis, but the methods and concepts we have developed are of potential use in a variety of other applications such as the monitoring of distributed power networks.

The approach we have developed highlights the major issues that must be addressed in designing distributed estimators, namely the aggregated modeling of the interactions between other portions of the overall process and the particular subprocess being estimated and the dynamic modeling of the information provided by other estimators as part of the process of producing coordinated, consistent estimates of all the subprocesses. We have presented systematic procedures for constructing these models that can in fact be used as the basis for a completely automated estimator design procedure [Doerschuk 1985].

In order to illustrate the various elements of our design process we have presented a case study corresponding to the tracking of a particular cardiac rhythm using synthetic data. The results presented indicate the potential of this design method. Two major issues remain to be considered, however, before a complete ECG rhythm analysis system can be constructed. In particular, while our distributed design yields estimators with far more modest computational demands than the corresponding optimal estimator, several steps can be taken to simplify these computations even more. First, as mentioned previously, it is possible to construct pipelined versions of our multi-pass estimators in which all passes are performed at the same time rather than in sequence. This achieves a several-fold increase in processing throughput. Also, the nature of the models arising in ECG analysis offer another possibility for simplification. Specifically, these finite-state processes typically display multiple time scale behavior (as actual signature-initiating events occur at a far lower rate than the sampling rate needed to capture interprocess timing). Consequently, it may be possible to use results on

hierarchical aggregation of processes with several time scales [Coderch 1983] to construct more efficient estimators that not only display the spatial decomposition we have exploited here but also the time scale decomposition of these processes.

Finally, it is important to realize that the problem of rhythm tracking addressed here is only a first step in a rhythm diagnosis sytems. Specifically in such a system one wishes to identify the underlying distributed process model from a set of such models representing different cardiac rhythms. As in standard system identification problems, the computation of the likelihoods for a set of models can be performed efficiently using the estimates produced by estimators based on each of the models (e.g. see [Gustafson 1978a, b] for an application of this idea to ECG rhythm analysis based on R-wave location data only). Doerschuk (1985) describes an approach to constructing such likelihoods based on the outputs of a set of estimators of the type described in this paper, but work remains to be performed to test this method and to develop efficient implementations.

Acknowledgements

We are grateful to Prof. R.G. Mark for the opportunity to use the M.I.T. Biomedical Engineering Center's computational facility.

References

[Ciocloda 1983]
G.H. Ciocloda, "Digital Analysis of RR Intervals for Identification of Cardiac Arrhythmias," Int. J. Bio-Medical Computing, Vol. 14, pp. 155-169, 1983.

[Coderch 1983]
M. Coderch, A.S. Willsky, S.S. Sastry, and D.A. Castanon, "Hierarchical Aggregation of Singularly Perturbed Finite State Markov Processes," Stochastics, Vol. 8, pp. 259-289, 1983.

[Doerschuk 1985]
P.C. Doerschuk, "A Markov Chain Approach to Electrocardiogram Modeling and Analysis," Ph.D. thesis, M.I.T. Dept. of Elec. Eng. and Comp. Sci.; also M.I.T. Lab. for Inf. and Dec. Sys. Rept. LIDS-TH-1452, April 1985.

[Doerschuk 1986]
P.C. Doerschuk, R.R. Tenney, and A.S. Willsky, "Modeling Electrocardiograms Using Interacting Markov Chains," submitted for publication; also Lab. for Inf. and Dec. Sys. Rept. LIDS-P-1491.

[Forney 1973]
G.D. Forney Jr., "The Viterbi Algorithm", Proc. IEEE, Vol. 61, No. 3, pp. 268-278, March, 1973.

[Gersch 1970]
W. Gersch, D.M. Eddy, and E. Dong Jr., "Cardiac Arrhythmia Classification: A Heart-Beat Interval-Markov Chain Approach", Comput. and Biomed. Res. (USA), Vol. 4, pp. 385-392, 1970.

[Gersch 1975]
W. Gersch, P. Lilly, and E. Dong Jr., "PVC Detection by the Heart-Beat Interval Data-Markov Chain Approach", Comput. and Biomed. Res. (USA), Vol. 8, pp. 370-378, 1975.

[Grove 1978]
T.M. Grove, V.K. Murthy, G.A. Harvey, and L.J. Haywood, "Comparison of R-R Interval Prediction Models", Med. Instrum. (USA), Vol. 12, No. 5, pp. 293-295, Sept.-Oct., 1978.

[Gustafson 1978a]
D.E. Gustafson, A.S. Willsky, J.-Y. Wang, M.C. Lancaster, and J.H. Triebwasser, "ECG/VCG Rhythm Diagnosis Using Statistical Signal Analysis, Part I: Identification of Persistent Rhythms", IEEE Trans. on Biomed. Eng., Vol. BME-25, No. 4, pp. 344-353, July, 1978.

[Gustafson 1978b]
D.E. Gustafson, A.S. Willsky, J.-Y. Wang, M.C. Lancaster, and J.H. Triebwasser, "ECG/VCG Rhythm Diagnosis Using Statistical Signal Analysis, Part II: Identification of Transient Rhythms", IEEE Trans. on Biomed. Eng., Vol. BME-25, No. 4, pp. 353-361, July 1978.

[Gustafson 1981]
D.E. Gustafson, J.-Y. Wang, and A.S. Willsky, "Cardiac Rhythm Interpretation Using Statistical P and R Wave Analysis", Frontiers of Engineering in Health Care, Houston, Sept., 1981.

[Haywood 1977]
L.J. Haywood, V.K. Murthy, G.A. Harvey, and T.M. Grove, "Comparison of Models for R-R Interval Prediction in ECG Monitoring", Proc. of the First Annual Symp. of Computer Applications in Medical Care, IEEE Computer Society, Washington, D.C., Oct. 1977, pp. 281-283.

[Richardson 1976]
J.M. Richardson, V.K. Murthy, and L.J. Haywood, "Nonlinear Markoff Models for Electrocardiographic R-R Interval Analysis", Math. Biosci. (USA), Vol. 29, pp. 299-307, 1976.

[Tsui 1975]
E.T. Tsui and E. Wong, "Sequential Approach to Heart-Beat Rhythm Classification", IEEE Trans. on Info. Theory, Vol. IT-21, No. 5, pp. 596-599, Sept., 1975.

END

12-87

DTIC